# BASH Lab 4 Flow Control

You can do this lab by using the techniques shown in the second script on slide 8 from the Cyber Aces Flow Control module.

Write a script that processes the output of `ls -l` and displays the name and size of files (not directories) that are larger than 100 in size. For example, if ls -l shows this:

[john@localhost ~]$ ls -l

total 56

drwxr-xr-x. 2 john john 4096 Sep 11 15:16 Desktop

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Documents

drwxr-xr-x. 3 john john 4096 Nov 14 04:49 Downloads

-rw-rw-r--. 1 john john 79 Oct 2 17:35 hash

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Music

drwxrwxr-x. 4 john john 4096 Sep 11 15:53 nc110

-rwxrwxr-x. 1 john john 148 Nov 19 02:43 params.sh

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Pictures

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Public

drwxrwxr-x. 2 john john 4096 Sep 22 04:48 svgs

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Templates

-rw-rw-r--. 1 john john 115 Nov 10 03:07 test.php

-rw-rw-r--. 1 john john 116 Nov 10 03:01 test.php~

drwxr-xr-x. 2 john john 4096 Sep 11 15:15 Videos

Your script should show this

./largefile

Name: params.sh Length 148

Name: test.php Length 148

Name: test.php~ Length 148

Hint: When writing scripts, it’s often easier to write them one step at a time. In this case:

1. We’re building around ls -l, so run that to see how the output looks.
2. We want to exclude directories, so pipe your output into a regex that looks for a dash at the beginning of the line”. Test/fix that until it works.
   1. ls –l | grep “your regex”
3. Now, try to grab the file name by adding awk to the pipeline
   1. ls –l | grep “your regex” | awk ‘{print $yourColumnNumber}’
4. Then repeat step 3, except change the column number to grab the file size/length instead
5. Now that you have the pieces working, put them into the format of the second script on slide 9 to make it set the NAME and LENGTH of the output it gets from ls –l. Add a line, echo “Name: $NAME Length: $LENGTH” so you can see what it did. Don’t worry about checking the file length yet.
6. Finally, add an if fi statement that checks for the file length longer than 100

Modify your script so that you can tell it the size limit as an argument. For example,

./largefile 50 would show this:

NAME: hash LENGTH: 79

NAME: params.sh LENGTH: 148

NAME: test.php LENGTH: 115

NAME: test.php~ LENGTH: 116

./largefile 120 would show this:

NAME: params.sh LENGTH: 148

# Hand In

Hand in the last script you wrote, that prints the length and name of files over the size limit you give as an argument to the command.